**B.S ANANGPURIA INSTITUTE OF TECHNOLOGY**

**AND MANAGEMENT**



**Object Oriented Programming**

**Lab File**

**Submitted To:                                                         Submitted By:**

**Ms Jolly Khurana Brijesh Kumar Sharma**

**PRATICAL NO:1**

**AIM :W**rite a java programme that prints all real solutions to the quadratic equations ax2+bx+c=0 . Read in a, b, c and use quadratic formula .if the discriminate is b2-4ac is negative , display a message stating that there are no real solutions

**SOURCE CODE:**

**import** java.util.Scanner;

**publicclass** Quadraticsolu

{

**publicstaticvoid** main(String args[])

{

**double**a,b,c;

**double**root1,root2;

System.***out***.println ("Enter the coefficient-a,b,c:");

Scanner sc=**new** Scanner (System.***in***);

a=sc.nextInt ();

b=sc.nextInt ();

c=sc.nextInt ();

**Double**d=b\*b-4\*a\*c;

**if** (d>0)

{

System.***out***.println ("The roots are real and different");

root1=(-b+Math.*sqrt*(d))/(2\*a);

root2=(-b-Math.*sqrt*(d))/(2\*a);

System.***out***.println ("root1="+root1+"and root2="+root2);

}

**elseif**(d==0)

{

System.***out***.println ("The roots are real and same");

root1=root2=(-b)/(2\*a);

System.***out***.println ("root1=root2="+root1);

}

**else**

{

System.***out***.println ("The roots are imaginary no real solution");

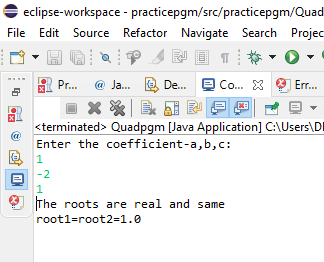
root1=root2=(-b)/(2\*a);

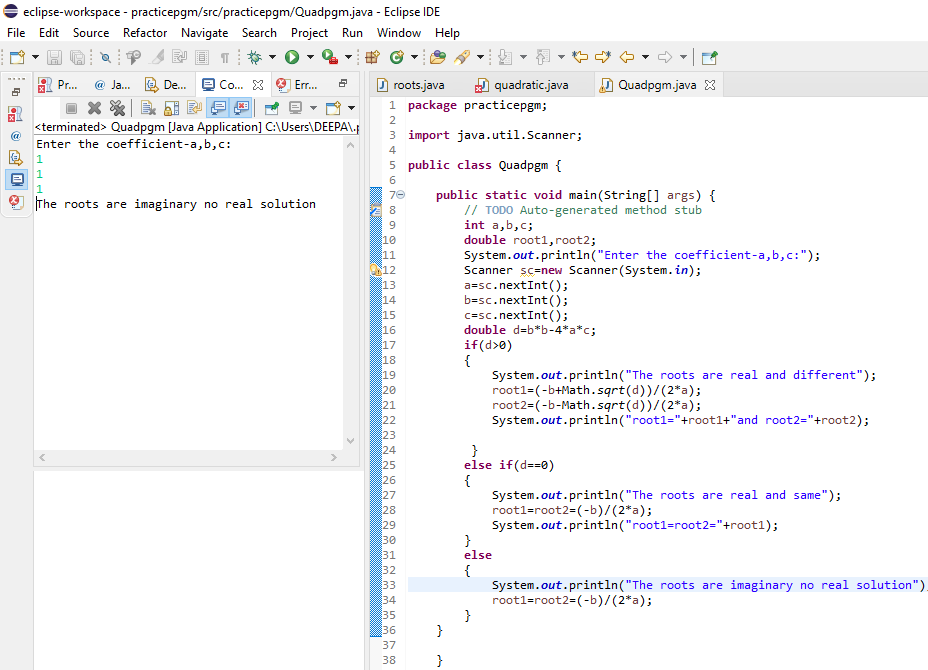
}

}

}

OUTPUT :





PRACTICAL NO : 2

AIM :- Write a program to find the average of given numbers using array.

SOURCE CODE :

Import java.util.scanner;

Public class Arraypgm{

Public static void main(string args[])

{

System.out.println(“how many no you want to enter “);

Scanner sc=new Scanner(System .in);

int n=sc.nextInt();

double[] arr=new double[n];

double total =0;

for(int i=0;i<arr.length;i++)

{

System.out.println(“enter element no.”+(i+1)+”:”);

arr[i] = sc.nextInt();

}

sc.close();

for(int i=0; i<arr.length ; i++);

{

total = total +arr[i];

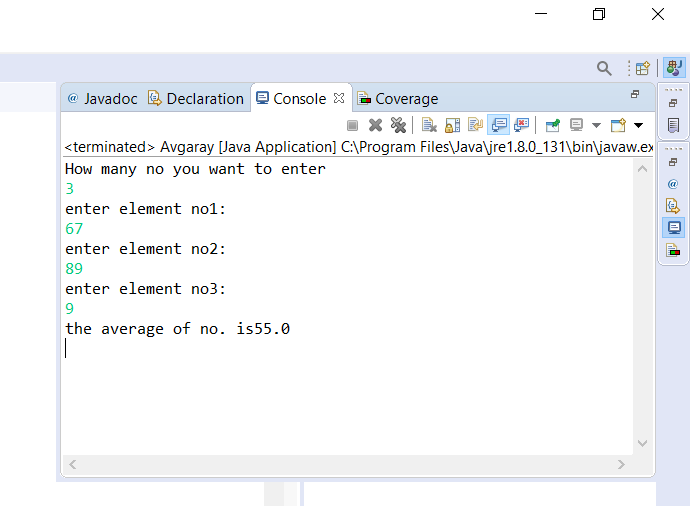
}

Double avg =total/arr.length ;

System.out.println (“the average of no.is “ +avg);

}

OUTPUT :



**PRACTICAL NO : 3**

# AIM : Write a program to print the pattern of numbers using multi-dimentional

Array -123

456

789

**SOURCE CODE :**

public class ArrM{

public static void main main(string args[])

{

int arr[] [] ={{1,2,3},{4,5,6},{7,8,9}};

for (int i=0 ;i<3 ;i++)

{

for (int j=0; j<3 ; j++)

{

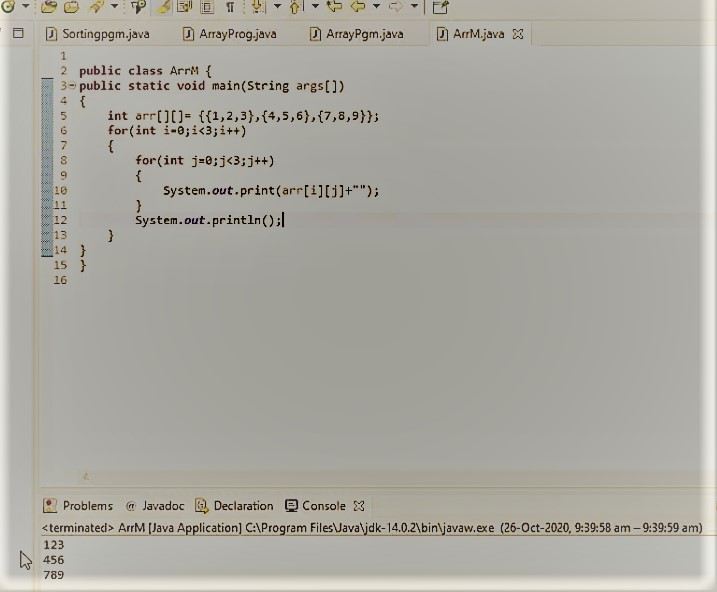
System.out .println(arr[i][j] + “ “);

` }

System.out .println();

}}}

OUTPUT :



PRACTICAL NO : 4

AIM : write a program to reverse a string in java .

Source Code :

Import java.util.Scanner ;

Public class StringRev {

Public static void main (String args []);

{

String name , rev = “ “c;

Scanner sc = new Scanner (System.in);

System.out.println (“Enter a string : “);

Name = sc.nextLine ();

int len = name.length ();

for (int i=len-1 ; i>=0; i--)

{

Rev=rev+name.charAt(i);

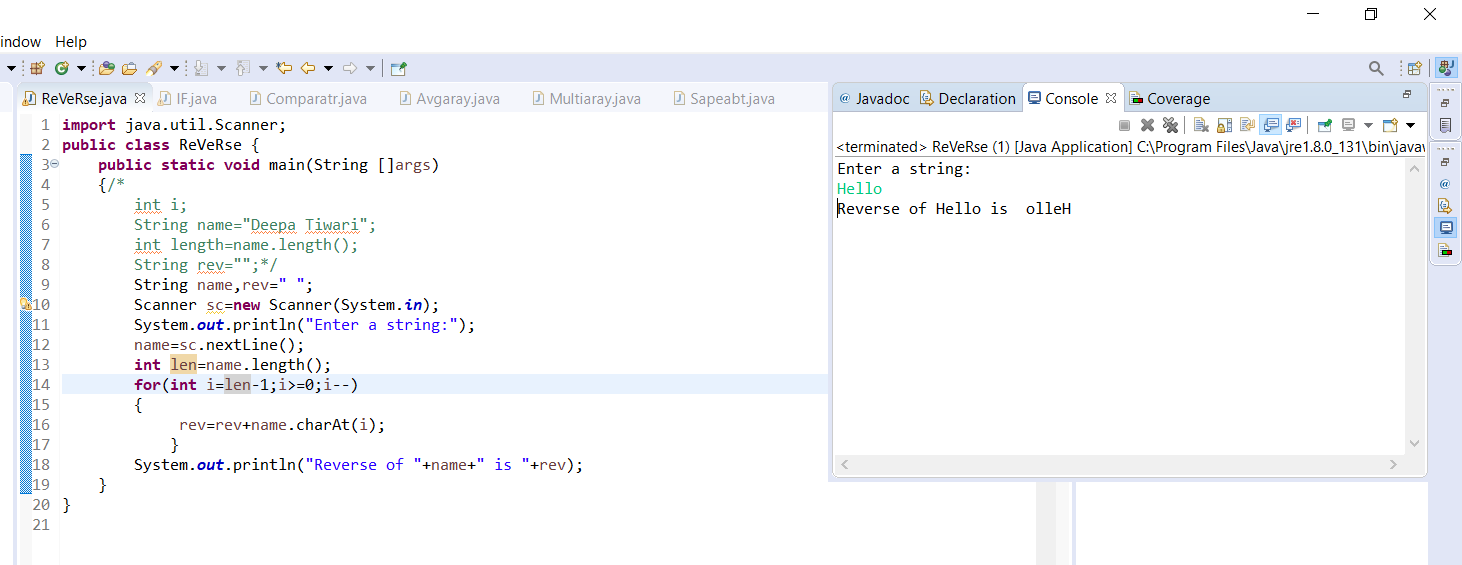
}

System.out.println (“Reversed String : “+rev) ;

}

}

OUTPUT:



PRACTICAL NO: 5

**AIM:** Write a program to create an abstract class named shape that contain two integer and an empty method name printarea () . Provide three class Rectangle,Triangle,and Circle such that each one of the classes extends the class shape . Each one of the class contain only the method printarea() that prints the area of the given shape.

**SOURCE CODE :**

Abstract class Shape

{

Public int x,y ;

Public abstract void printarea() ;

}

Class Rectangle extends Shape

{

Public void printarea()

{

System.out.println(“ Area of Rectangle is : “+x\*y);

}

}

Class Triangle extends Shape

{

Public void printarea();

{

System.out.println(“Area of Triangle :”+(x\*y)/2);

}

}

Class Circle extends Shape

{

Public void printarea()

{

System.out.println(“Area of Circle is : “+(3.14\*x\*x));

` }

}

Public class ShapeAbt {

Public static void main (String args[])

{

Rectangle r=new Rectangle();

r.x=10;

r.y=20;

r.printarea();

Triangle t=new triangle();

t.x=30;

t.y=35;

t.printarea();

Circle c=new Circle();

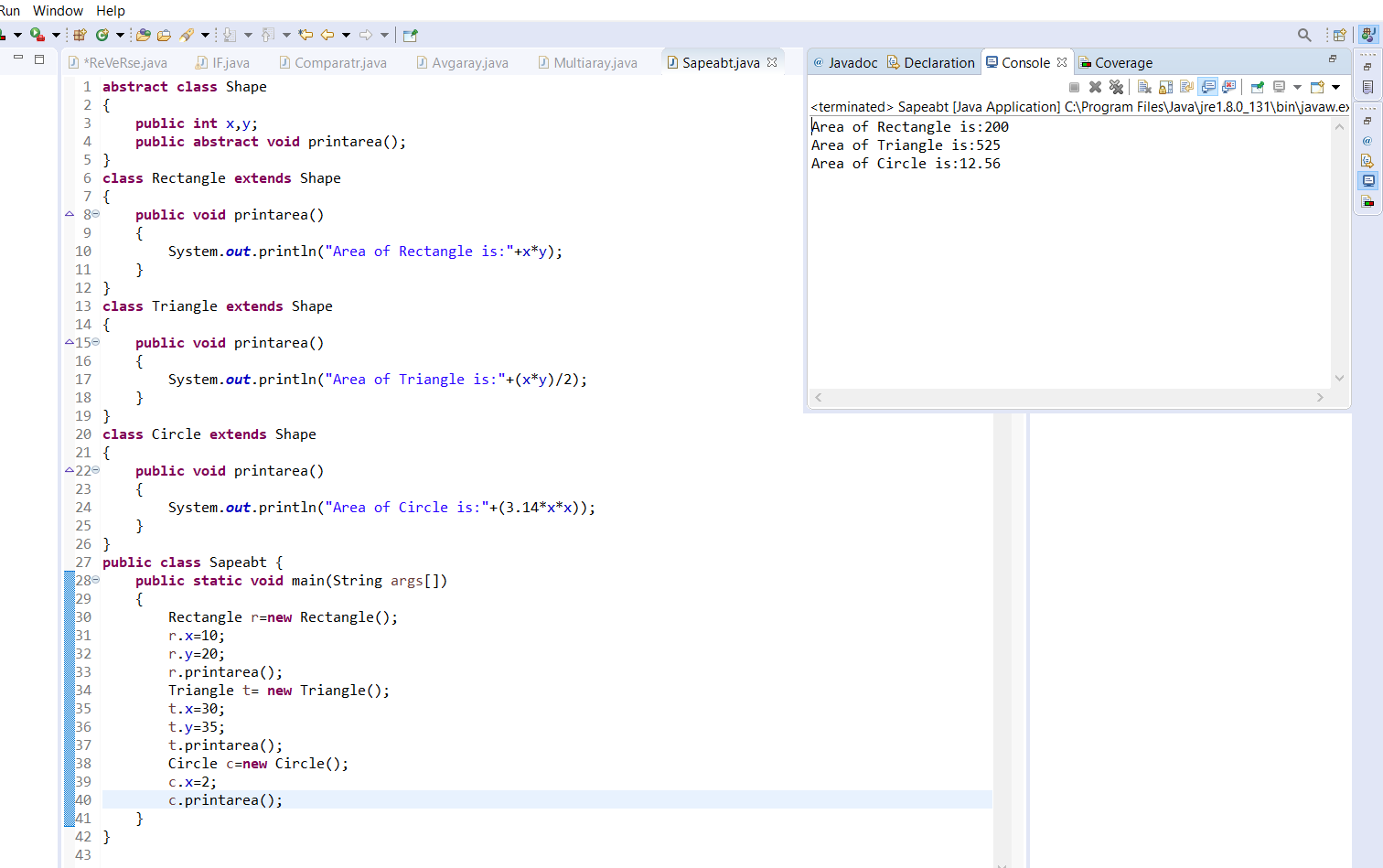
c.x=2;

c.printarea();

}

}

**OUTPUT:**

****

**PRACTICAL NO: 6**

**AIM : Write a java program to design a class to represent a bank account which contain Account No ,Balance amount in the bank account .Use constructor to assign initial values , to deposit an amount , to withdraw amount after checking balance ,to display name and balance .**

**SOURCE CODE :**

**import** java.util.Scanner;

**class** BankAccount

{

**int**Accountno;

String name;

String type;

**int**Balance;

**void** BankAccount(**int**a, String b, String c,**int**d)

{

**this**.Accountno=a;

**this**.name=b;

**this**.type=c;

**this**.Balance=d;

}

**void** deposite(**int**s)

{

System.***out***.println("Balance before deposite"+Balance);

Balance=Balance+s;

System.***out***.println("Balance after deposite"+Balance);

}

**void** withdraw(**int**w)

{

System.***out***.println("Balance before withdraw"+Balance);

**if**(w>Balance || Balance<=0)

{

System.***out***.println("You can not withdraw any amount");

}

**else**

Balance=Balance-w;

System.***out***.println("Balance after withdraw"+Balance);

}

**void** display()

{

System.***out***.println("Name is:"+name);

System.***out***.println("Balance is:"+Balance);

}

}

**publicclass** Bank {

**publicstaticvoid**main(String[] args) {

// **TODO** Auto-generated method stub

BankAccount ba=**new** BankAccount();

Scanner sc=**new** Scanner (System.***in***);

System.***out***.println("Enter AccountNo,Name,Type ,Balance");

ba.BankAccount(sc.nextInt(),sc.next(), sc.next(),sc.nextInt());

System.***out***.println("Enter the amount to deposite:");

ba.deposite(sc.nextInt());

System.***out***.println("Enter the amount to withdraw:");

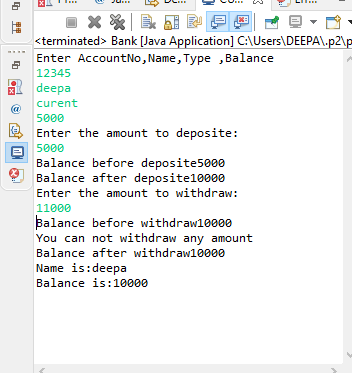
ba.withdraw(sc.nextInt());

ba.display();

}

}

OUTPUT :



**PRACTICAL NO: 7**

**AIM : Write a program to create user interface to perform integer division .**

**The user enters two numbers in the text field Num1, Num2. The division Of these two numbers is displayed in the result field when the division button is clicked. if num1,num2 were not integer ,the program would throw a NumberFormatException ,if num2 is zero and the program would throw an ArithmaticException Display the exception in the message box .**

**SOURCE CODE :**

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**import**javax.swing.event.\*;

**class**A**extends** JFrame **implements** ActionListener

{

JLabel l1,l2,l3;

JTextField tf1,tf2,tf3;

JButton b1;

A()

{

setDefaultCloseOperation (JFrame.***EXIT\_ON\_CLOSE***);

setLayout(**new** FlowLayout());

l1=**new** JLabel ("Enter Num1");

add(l1);

tf1=**new** JTextField(10);

add(tf1);

l2=**new** JLabel("Enter Num2");

add(l2);

tf2=**new** JTextField(10);

add(tf2);

l3=**new** JLabel("Result");

add(l3);

tf3=**new** JTextField(10);

add(tf3);

b1=**new** JButton ("Divide");

add(b1);

b1.addActionListener(**this**);

setVisible(**true**);

}

**publicvoid** actionPerformed(ActionEvent ae)

{

**try**

{

**int**a=Integer.*parseInt*(tf1.getText());

**int**b=Integer.*parseInt*(tf2.getText());

**if**(b==0)

**thrownew** ArithmeticException("Divide by Zero");

**float**c=a/b;

tf3.setText(String.*valueOf*(c));

}

**catch**(NumberFormatException ex)

{

JOptionPane.*showMessageDialog*(**this**,ex.getMessage());

}

**catch**(ArithmeticException ex)

{

JOptionPane.*showMessageDialog*(**this**,ex.getMessage());

}

}}

**publicclass** interfacepgm {

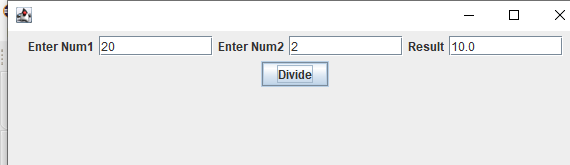
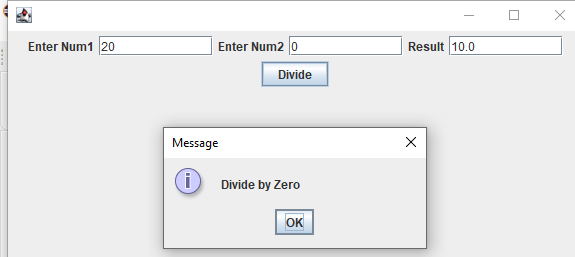
**publicstaticvoid** main(String args[])

{

A a=**new**A();

}}

OUTPUT:



**PRACTICAL NO: 8**

**AIM : Write a program to perform Multiple inheritance in java .**

**SOURCE CODE :**

**interface** vehicle1

{

**int*speed***=30;

**publicvoid** distance();

}

**interface** vehicle2

{

**int*distance***=60;

**publicvoid** speed();

}

**class** vehicle **implements** vehicle1 , vehicle2

{

**publicvoid** distance ()

{

**int**distance =***speed***\*100;

System.***out***.println("Distance travelled is :" +distance);

}

**publicvoid** speed ()

{

**int**speed=***distance***/100;

System.***out***.println("Speed is:"+speed);

}

}

**publicclass** multipleinhert {

**publicstaticvoid** main(String[] args) {

vehicle v=**new** vehicle();

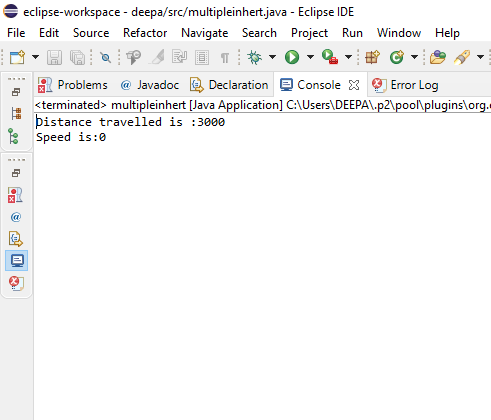
v.distance();

v.speed();

}

}

OUTPUT :



**PRACTICAL NO = 9**

**AIM: Write a program to perform multithreading in java .**

**SOURCE CODE :**

**publicclass** multithread {

**publicstaticvoid** main(String args[])

{

B b1=**new**B("hello");

b1.start ();

B b2=**new**B("hi");

b2.start ();

}

}

**class** B **implements** Runnable

{

Thread t1;

**private** String name;

B (String n)

{

name=n;

}

**publicvoid** run()

{

System.***out***.println ("Thread running:"+name);

**for**(**int**i=0;i<4;i++)

{

System.***out***.println (i);

System.***out***.println(name);

**try**

{

Thread.*sleep*(1000);

}

**catch**(InterruptedException e)

{

System.***out***.println("Thread has been interrupted");

}

}

}

**publicvoid** start()

{

System.***out***.println("Thread Started");

**if**(t1==**null**)

{

t1=**new** Thread(**this**,name);

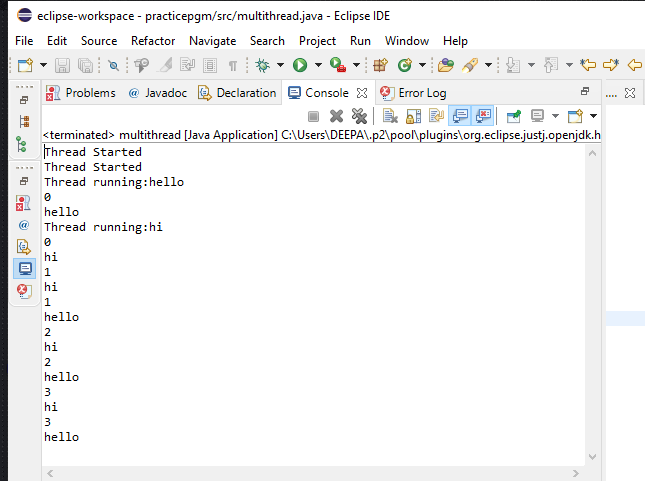
t1.start();

}

}

}

OUTPUT:



PRACTICAL NO: 10

AIM – Write a program to perform comparator in java .

SOURCE CODE-

1. Import java .util.\*;
2. Class student
3. {
4. int rollno;
5. String name ,address ;
6. Public student(int rollno ,string name,string address )
7. {
8. this . rollno = rollno;
9. this . name =name;
10. this.address = address;
11. }
12. public string to string()
13. {
14. return this.rollno + “ “ + this.name + “ “ + this.address ;
15. }
16. }
17. class sortedbyrollno implements comparator < student >
18. {
19. public int compare (student a, student b)
20. {
21. return a. rollno – b.rollno ;
22. //if a < b , then a – b will be negative ;
23. //if a > b , then a –b will be positive ;
24. }
25. }
26. class sortedbyname implements comparator < student >
27. {
28. public int compare ( student a , student b )
29. {
30. return a.name.compareTo(b.name) ;
31. }
32. }
33. class Sortingpgm
34. {
35. public static void main (string[] args )
36. {
37. ArrayList<student> ar = new ArrayList<student>();
38. ar.add (new student(111,”bbbb”,”London”));
39. ar.add (new student(131,”aaaa”,”nyc”));
40. ar.add (new student(121,”cccc”,”jaipur”));
42. System .out . println (“unsorted “);
43. for ( int i=0; i< ar , . size (); i++)
44. System.out.println (ar.get (i));
45. collections.sort (ar , new Sortbyroll());
46. system.out.println (“ \n sorted by rollno “);
47. for(int i=0 ; i<ar .size ();i++ )
48. System.out. println ( ar.get(i));
49. collection.sort (ar,new sortbyname());
50. System.out.println (“\ n sorted by name “);
51. for (int i=0 ; i<ar.size () ; i++)
52. System.out.println (ar.get(i));
53. }
54. }

output:

